Handwritten Music Object Detection: Open Issues and Baseline Results

Alexander Pacha, Kwon-Young Choi, Bertrand B. Coüasnon, Yann Ricquebourg, Richard Zanibbi, Horst Eidenberger

Handwritten Music Object Detection: Open Issues and Baseline Results

Alexander Pacha*, Kwon-Young Choi†, Bertrand Coüasnon†,
Yann Ricquebourg†, Richard Zanibbi‡ and Horst Eidenberger*
*Institute for Software Technology and Interactive Systems, TU Wien, Vienna, Austria
Email: *alexander.pacha@tuwien.ac.at*
†Univ Rennes, CNRS, IRISA, F-35000 Rennes, France
Email: *kwon-young.choi@irisa.fr, bertrand.couasnon@irisa.fr, yann.ricquebourg@irisa.fr*
‡Rochester Institute of Technology, Rochester, USA
Email: *rlaz@cs.rit.edu*

***Abstract*—Optical Music Recognition (OMR) is the challenge of understanding the content of musical scores. Accurate detection of individual music objects is a critical step in processing musical documents, because a failure at this stage corrupts any further processing. So far, all proposed methods were either limited to typeset music scores or were built to detect only a subset of the available classes of music symbols. In this work, we propose an end-to-end trainable object detector for music symbols that is capable of detecting almost the full vocabulary of modern music notation in handwritten music scores. By training deep convolutional neural networks on the recently released MUSCIMA++ dataset which has symbol-level annotations, we show that a machine learning approach can be used to accurately detect music objects with a mean average precision of up to 80%.**

***Keywords*-Optical Music Recognition; Object Detection; Handwritten Scores; Deep Learning**

I. Introduction

Optical Music Recognition (OMR) attempts to understand the musical content of documents containing printed or handwritten music scores by recognizing the visual structure and the objects within a music sheet. Once, all objects are recognized, a semantic reconstruction step attempts to understand the relations of objects to each other and recover the musical semantics. With recent advances in computer vision, accelerated by the popularity of deep convolutional neural networks (CNN), OMR received a number of groundbreaking contributions that generate very accurate results for particular sub-problems, such as staff line removal [1] or symbol classification [2]. In this work, we investigate the challenge of music object detection which aims at accurately detecting music objects in music scores. Music objects can be both primitive glyphs (e.g. note-head, stem, beam) or compound symbols (e.g. notes, key-signatures, time-signatures) used in music notation. A music object detector takes an image and outputs the bounding-box and class-label for each found object. Traditionally, this was solved by first removing the staff lines, followed by symbol segmentation and classification (see Figure 1) [3].

In this work, we present the first attempt to establish a baseline for music object detection of handwritten scores with the full vocabulary of modern music notation. By following a machine learning approach and using an end-to-end trainable object detector on the recently published MUSCIMA++ dataset, we demonstrate how to build a generalizable and accurate music object detector and investigate the effects of various technical choices like the use of a particular detector or feature extractor.

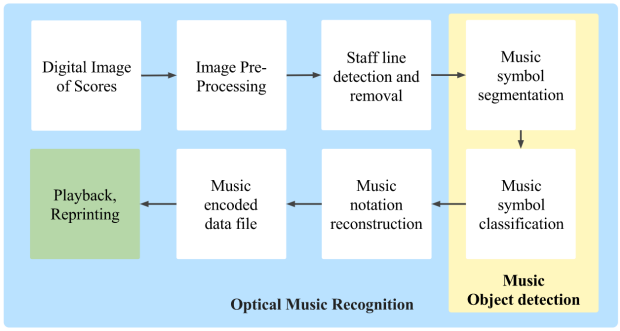

Figure 1. The traditional pipeline for Optical Music Recognition. Music object detection subsumes segmentation and classification of music symbols.

II. Related Work

Visual object detection is a very active field of research with remarkable results on detecting objects in natural images with a variety of active competitions. Many competing approaches have been proposed in the last few years such as Faster R-CNN [4], R-FCN [5] and Single shot detectors [6], [7]. While some optimize for accuracy, others strive for high performance [8]. However, all of them share the fact, that heavily make use of deep convolutional neural networks.

The traditional pipeline of segmenting and classifying symbols has been shown to work well on simple typeset music scores with a known music font [9]. But when considering low-quality images, complex scores or even handwritten ones, these systems tend to fail, mainly because errors propagate from one step to subsequent steps [10], e.g. a segmentation error could cause incorrectly detected objects. Initial attempts to overcome this limitation by directly detecting music objects with CNNs were made by [11], who suggest an adaptation of Faster R-CNN with a custom region proposal mechanism based on the morphological skeleton to accurately detect noteheads and [12], who are able to detect accidentals in dense piano

scores with high accuracy, given previously detected noteheads, that are being used as input-feature to the network. However, both of them are limited to experimentations on a tiny subset of the full vocabulary used in modern music notation. Although both approaches can be extended to other classes, it remains an open question, whether a general purpose detector that can learn a large vocabulary is superior to multiple class-specific detectors.

A very interesting alternative to the traditional OMR pipeline is the attempt of solving OMR in a holistic fashion. The first notable attempt at doing so was by Pugin [13], who used Hidden Markov Models to read typographic prints of early music. More recently, the combination of using CNNs jointly with Recurrent Neural Networks to build an end-to-end trainable OMR system [14] was adapted and extended by [15] and [16]. Both train very similar models on a very large set of monophonic music scores containing a single staff per image. Although the reported results on the given datasets are very good, these systems currently exhibit the following limitations:

- They operate only on very primitive, printed, monophonic scores. Extending their pipeline to more complex music scores with multiple voices requires a different formulation of the output data to at least include onset and offset of each note and not only the pitch and duration.
- By using pooling operations during the feature extraction, the network gains location invariance that conflicts with the interest of precise location information, which is needed to correctly infer the pitch of a note.
- By omitting the positional information of individual symbols and only considering the audible information of music symbols as output, such systems restrict themselves to replayability, as reprinting of music scores requires precise positional information [17].

While in theory semantic segmentation of the scores would go one step further and extract considerable more information - basically a classification of each pixel - two things should be noted: classifying pixels assumes that the class of each pixel is unique and mutually exclusive [18] - an assumption that might not hold for overlapping symbols but can probably be ignored for practical applications; and most traditional systems that attempt to perform semantic reconstruction operate on detected objects, not on individual pixels, thus requiring a clustering step after the semantic segmentation. Therefore we argue that detecting bounding boxes of musical objects is sufficient for performing OMR.

III. The Challenge of Detecting Music Symbols

When comparing music object detection to detection of objects in natural scenes or optical character recognition, two unique challenges are worth noting: firstly, music scores often have a very high density of objects with more than 1000 objects printed on a single page. Secondly, the relative position between a symbol and its staff lines is crucial. Already a tiny error along the y-axis may have a significant impact on recovering the correct pitch of a note.

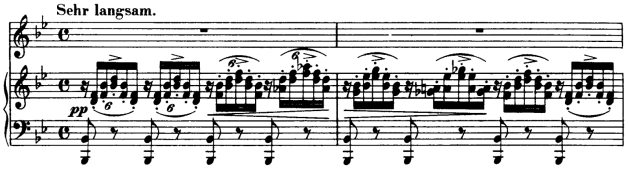

Figure 2. Facsimile of Franz Schuberts Ave Maria D. 839, with simplifications in the second bar that intentionally violates syntactic rules of common music notation.

The detection of music objects is of paramount importance to the overall OMR process because once all symbols were detected accurately, a set of rules can be applied to infer the semantics of the objects and perform music notation reconstruction as demonstrated by [19]. We also suggest that the point right after individual objects were detected and classified, is probably the best moment for putting the user into the loop, if that is intended. Fixing errors at this stage can be performed locally without dealing with complicated semantic rules or affecting neighboring symbols (changing the duration of a single note in a music notation program often entails side-effects on other notes within the same of subsequent bars). Highlighting uncertain detections and suggesting likely alternatives could improve the usability and reduce editing costs even further.

Note that even with all symbols being correctly detected and classified, recovering the musical semantics still remains a very challenging problem, as demonstrated in Figure 2. Here, the second staff in the first bar contains a small 6 for each tuplet, indicating that the first rest and the following five chords sum up to a quarter note. This small number is intentionally omitted in the second bar for simplification but would now result in an invalid meter if interpreted in isolation. Only with the preceding information and prior knowledge about common simplifications, a musician can interpret the scores correctly.

To be able to introduce such semantic in an OMR system, it is necessary to formalize and use musical notation knowledge. Rule-based system can perform such formalization. For example, with the DMOS system [19] it has been possible to formalize the musical notation, graphically, syntactically and sometimes semantically, for full polyphonic scores, and produce a system which allows to split notes into voices and use the vertical alignments of synchronized notes in orchestral scores as well as the number of beats in a bar to detect and correct recognition errors. This grammatical formalization is built on terminals which correspond to the musical objects we propose to recognize with deep convolutional neural network.

IV. Building a Music Object Detector

For building a robust and extensible music object detector, we propose a machine-learning approach by using deep convolutional neural networks that operate directly

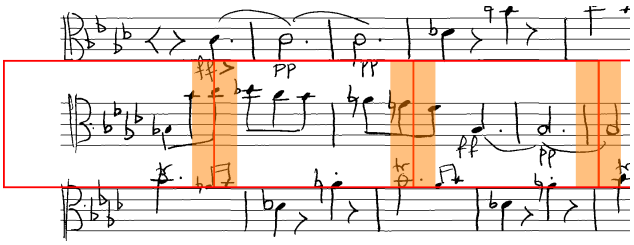

Figure 3. Crop regions (red) for extracting meaningful sub-images with horizontally overlapping areas (orange).

on the input image. This simplifies the OMR process to the following steps: preprocessing, music object detection, and semantic reconstruction. Steps such as removing the staff lines and segmenting symbols do not need to be addressed explicitly. Existing state-of-the-art object detectors such as Faster R-CNN or R-FCN were designed to detect objects in natural scenes and often fine-tuned to work well on publicly available datasets such as COCO [20] or ImageNet [21]. Applying them out-of-the-box on a different dataset with many densely packed objects could lead to sub-optimal performance. Therefore we suggest applying a certain amount of preprocessing to the data and tailor these detectors to perform well on the task at hand.

A. Dataset

For training a music object detector, we use the MUSCIMA++ dataset [22], as it contains 140 high-quality images with over 90000 symbol-level annotations made by human annotators across 105 different classes of music symbols for the underlying CVC MUSCIMA dataset [23]. The images have a high resolution of about 3500x2000 pixel, are binarized and optionally come with staff-lines removed. To efficiently train an object detector on such images, the image size has to be reduced. We propose to crop the images in a context-sensitive way, by cutting images first vertically, such that each image contains only one staff and then horizontally to have a width-to-height-ratio of no more than 2:1. To do so, we manually determine the approximately 200 vertical slices in a way that the staff is fully included along with all objects that belong to it, before horizontally cutting the images with about 15% overlap to adjacent slices (see Figure 3). Basically, each horizontal slice extends from the bottom of the staff above to the top of the staff below. This cropping can also be done by automatically detecting staffs and then applying the same slicing-rules leading to image crops that partially overlap both horizontally and vertically. One limitation of this approach is that objects, that significantly exceed the size of such a cropped regions will not appear in the ground truth, as only annotations that have an intersection-over-area of 0.8 or higher between the object and the cropped region will be included.

As music objects, we consider the full vocabulary of all 105 classes contained in the MUSCIMA++ dataset, containing both primitives such as note-heads as well as compound objects such as key-signatures that consist of one or multiple accidentals.

B. Experimental Design

For evaluating our suggested approach, we conducted several experiments to study the effects of different detectors, different feature-extractors, staff line removal, a reduced set of classes and transfer-learning. Using the deep learning library TensorFlow, we adapted the work from [8] to detect music objects by training on the data described in Section IV-A. The entire source-code, including training protocols and detailed instruction to reproduce our results can be found at http://github.com/apacha/MusicObjectDetector-TF. We considered:

- the three meta-architectures Faster R-CNN, R-FCN, and SSD as object detectors
- ResNet50, Inception-ResNet-v2, MobileNet-v1 and Inception-v2 as feature extractors, excluding custom-made networks that cannot benefit from transfer-learning
- images with and without staff lines (based on the images provided along the CVC-MUSCIMA dataset)
- the full vocabulary of all 105 classes included in the MUSCIMA++ dataset, as well as a reduced number of only 71 classes (named MUSCIMA++71), removing 34 classes that appear less than 50 times in the ground truth and are only of minor importance such as uncommon numerals and letters. Exceptions were only made for the classes double sharp and the numerals 5, 6, 7 and 8: although they appear less than 50 times in the dataset, we consider them essential to recover music semantics such as pitch and time signature.
- an even further reduced dataset (named MUSCIMA++49) with only 49 classes remaining, furthermore removing compound objects (like key signature and time signature) and linear objects (like beams, stems, measure separators and alike with strong variation of width-to-height ratio).
- initializing the training with random weights as well as pre-trained weights on the COCO dataset

All of the above-mentioned object detectors have a certain set of hyperparameters that need to be fine-tuned for the particular dataset. For example, [7] shows that using statistical analysis to obtain a sensitive number of anchor boxes, anchor box sizes and, ratios, can improve the results significantly, compared to hand-picked priors. When running similar analysis on the cropped images, we obtain the following characteristics: For a typical input image of 600 pixels width and 300 pixels height (see Figure 4), we found the average square box size is about 37 pixels with a standard deviation of 48 pixels. However, the dataset contains also extreme cases of small objects like dots with a size of few pixels and large objects with a size of hundreds of pixels. The mean ratio between width and height of boxes is of 0.7 which means that generally box heights are larger than their widths. Finally, cropped images that are to be fed to the detector contains

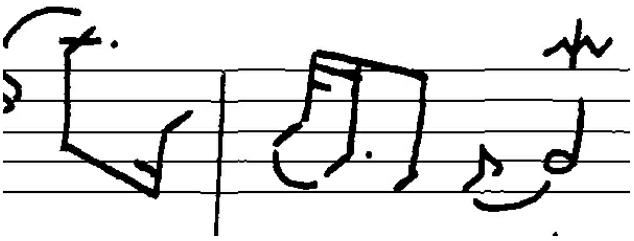

Figure 4. Typical sample of a cropped image that serves as input for the music object detector.

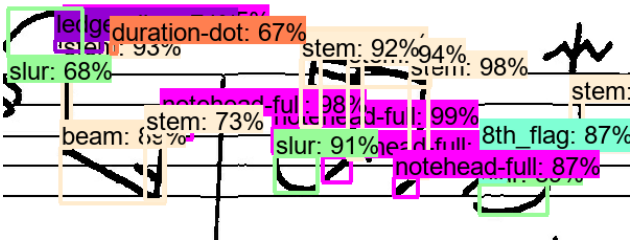

Figure 5. Typical detection results with most symbols recognized correctly.

an average of 19 symbols with a standard deviation of 11. Therefore we significantly shrunk the anchor boxes generated with Faster R-CNN and F-RCN to 8x8, 16x16, 32x32 and 64x64 pixels with aspect ratios of 1:2, 1:1 and 2:1 and a stride of 8 pixels.

C. Evaluation and Results

Following the evaluation protocols of the Pascal VOC challenge [24], we report the mean average precision (mAP) for each completed training in Table I and the detailed average precision per class for the combination that yielded the best results in Table II. Figure 5 shows a typical detection within a single image.

We find that the best performing detector is the Faster R-CNN using the Inception-Resnet V2 feature extractor pre-trained on the COCO dataset. This model produces a mAP of 80%. When comparing the results of training with and without staff lines, it seems like the removal of staff lines only has a minimal impact on the results, confirming the claim of [13] that staff line removal is no longer necessary for modern methods like deep learning. However, readers should also note that the staff lines in the MUSCIMA dataset are synthetic and do not experience the usual distortions that apply to scans or pictures of real music scores. Other detectors like the R-FCN or SSD produce good results as well, with a mAP of 75% and 62% respectively. Our results, therefore, comply with the findings of [8], where in particular the SSD model trades smaller accuracy for higher processing speed.

Modifying the set of classes by removing under-represented classes of less than 50 samples, boosted the mAP by 6% (MUSCIMA++71). Lots of music object are composed of straight and curved lines. Knowing that state-of-the-art line detection systems are now extremely accurate, we trained a model on a subset without line-shaped and under-represented classes (MUSCIMA++49). However, we found that in practice, the removal of these line-shaped classes does not impact the accuracy of other classes. Note, that in Table II, six classes did not have any fully visible instances in the test set because they did not fit within the cropping regions and were thus discarded, leading to undefined results.

V. Discussion and Conclusion

In this work, we show that state-of-the-art deep learning detectors like Faster R-CNN, R-FCN and SSD can produce accurate detection results on a wide range of music symbols. After exploring different feature extractors, transfer learning from the COCO dataset, images with and without staff lines and different sets of classes, we achieve a mAP of 80%, which is a decent baseline. However, there are still the following open issues that need to be addressed in future work.

The best way of processing a whole page of a score remains an open question. In this work, we used a simple overlapping sliding window approach. This method, although simple to use, has many well-known downsides like the poor performance of processing empty images or cutting up large symbols as well as a non-trivial merging step that has to fuse information from multiple overlapping sections. Lots of work in the literature of image processing tackle these problems with attention mechanism or region proposals, which should improve this pre-processing step.

Another problem, specific to OMR, is the inherent imbalance of symbol classes: some symbols like noteheads are extremely frequent whereas others like double sharps are rare and often tied to a specific type of score. Having experimented with state-of-the-art deep learning object detectors, we found that classes do not interact with each other: simplifying the task by removing line-shaped classes did not improve the overall precision. However, there is a minimum threshold of samples, around 20 samples, for each class in order to be meaningful during the training but this does not guarantee that the model is not overfitting. New work like the RetinaNet with a focus loss [25] could be a way to solve imbalance in class and improve the training, especially on hard to detect classes.

Although in this work, we used the test set proposed by MUSCIMA++, where writers in the test set do not appear in the training set, we are still not sure if this system is truly writer independent. One way to ensure this is to propose cross-validation test sets where we can evaluate every writer in the dataset independently.

Finally, we show that in our case, the removal of these staff lines is unnecessary. Future experiments that apply data-augmentation using noise models and deformed images as proposed for the staff removal challenge [26] can give more insights, whether removing staff lines is still needed in the OMR pipeline.

Acknowledgment

The authors would like to thank all creators of public OMR datasets for collecting them and making them available to other researchers.

Table I
DETAILED RESULTS FOR VARIOUS HYPER-PARAMETER COMBINATIONS OF THE MUSIC OBJECT DETECTOR.

Meta-Architecture	Feature Extractor	Pre-trained weights	Number of classes	Images have staff lines	Mean Average Precision on Test Set (%)	Weighted Mean Average Precision on Test Set (%)
Faster R-CNN	Inception-ResNet-v2	✓	105	✓	69.53	93.58
Faster R-CNN	Inception-ResNet-v2	✓	105	✗	70.98	93.58
Faster R-CNN	Inception-ResNet-v2	✓	49	✓	77.00	94.66
Faster R-CNN	**Inception-ResNet-v2**	✓	**71**	✓	**77.20**	**93.34**
Faster R-CNN	Inception-ResNet-v2	✓	71	✗	80.00	93.91
Faster R-CNN	ResNet50	✓	105	✓	67.99	92.74
R-FCN	ResNet50	✗	71	✓	65.36	86.37
R-FCN	ResNet50	✓	71	✓	63.02	87.38
R-FCN	ResNet50	✓	71	✗	75.24	92.32
R-FCN	Inception-ResNet-v2	✓	105	✓	50.73	84.12
R-FCN	Inception-ResNet-v2	✓	71	✓	63.05	86.13
SSD	Inception-v2	✓	71	✗	57.92	72.30
SSD	Inception-v2	✓	71	✓	57.44	69.27
SSD	Inception-v2	✓	105	✓	62.00	82.19

REFERENCES

[1] A.-J. Gallego and J. Calvo-Zaragoza, "Staff-line removal with selectional auto-encoders," *Expert Systems with Applications*, vol. 89, pp. 138 – 148, 2017.

[2] A. Pacha and H. Eidenberger, "Towards a universal music symbol classifier," in *Proceedings of the 12th IAPR International Workshop on Graphics Recognition*, IAPR TC10 (Technical Committee on Graphics Recognition). New York, USA: IEEE Computer Society, 2017, pp. 35–36.

[3] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. Marcal, C. Guedes, and J. S. Cardoso, "Optical music recognition: state-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, 2012.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[5] Y. Li, K. He, J. Sun *et al.*, "R-FCN: Object detection via region-based fully convolutional networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 379–387.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[7] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," *arXiv preprint arXiv:1612.08242*, 2016.

[8] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/accuracy trade-offs for modern convolutional object detectors," *CoRR*, vol. abs/1611.10012, 2016.

[9] F. Rossant and I. Bloch, "Robust and adaptive OMR system including fuzzy modeling, fusion of musical rules, and possible error detection," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, p. 081541, 2006.

[10] A. Pacha and H. Eidenberger, "Towards self-learning optical music recognition," in *Proceedings of the 16th IEEE International Conference On Machine Learning and Applications*, 2017, in print.

[11] J. j. Hajič and P. Pecina, "Detecting noteheads in handwritten scores with convnets and bounding box regression," *arXiv preprint arXiv:1708.01806*, 2017.

[12] K.-Y. Choi, B. Coüasnon, Y. Ricquebourg, and R. Zanibbi, "Bootstrapping samples of accidentals in dense piano scores for cnn-based detection," in *Proceedings of the 12th IAPR International Workshop on Graphics Recognition*, IAPR TC10 (Technical Committee on Graphics Recognition). New York, USA: IEEE Computer Society, 2017.

[13] L. Pugin, "Optical music recognitoin of early typographic prints using hidden markov models." in *ISMIR*, 2006, pp. 53–56.

[14] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2016.

[15] J. Calvo-Zaragoza, J. J. Valero-Mas, and A. Pertusa, "End-to-end optical music recognition using neural networks," in *18th International Society for Music Information Retrieval Conference*, 2017.

[16] E. van der Wel and K. Ullrich, "Optical music recognition with convolutional sequence-to-sequence models," *arXiv preprint arXiv:1707.04877*, 2017.

[17] H. Miyao and R. M. Haralick, "Format of ground truth data used in the evaluation of the results of an optical music recognition system," in *IAPR workshop on document analysis systems*, 2000, pp. 497–506.

[18] J. Calvo-Zaragoza, G. Vigliensoni, and I. Fujinaga, "A machine learning framework for the categorization of elements in images of musical documents," in *Third International Conference on Technologies for Music Notation and Representation. A Coruna: University of A Coruna*, 2017.

Table II
DETAILED PRECISION RESULTS PER CLASS FOR THE BEST OBTAINED MUSIC OBJECT DETECTOR THAT USED IMAGES WITH STAFF LINES (SEE TABLE I, LINE 4).

Class name	Number of samples	Average precision on the test set (%)
16th_flag	499	32.77
16th_rest	436	98.19
8th_flag	2200	94.05
8th_rest	1134	99.60
accent	201	95.68
beam	6593	93.00
c-clef	189	99.78
double_sharp	39	83.33
duration-dot	2067	91.19
dynamics_text	681	77.14
f-clef	284	99.78
flat	1112	98.61
g-clef	402	100.00
grace_strikethrough	74	74.83
grace-notehead-full	348	69.28
hairpin-cresc.	224	NaN
hairpin-decr.	276	NaN
half_rest	216	98.71
key_signature	695	75.97
ledger_line	6847	92.18
letter_a	84	63.66
letter_c	224	65.53
letter_d	164	88.09
letter_e	443	62.13
letter_f	508	89.04
letter_i	95	41.01
letter_l	87	62.31
letter_M	56	70.38
letter_m	142	30.66
letter_n	93	72.89
letter_o	326	82.00
letter_P	79	90.10
letter_p	510	28.47
letter_r	385	37.25
letter_s	329	69.43
letter_t	272	83.42
letter_u	50	7.26
measure_separator	2854	61.61
multi-staff_brace	113	NaN
multi-staff_bracket	51	NaN
multiple-note_tremolo	112	48.00
natural	1090	97.90
notehead-empty	1669	99.51
notehead-full	21333	99.89
numeral_2	34	80.26
numeral_3	327	95.63
numeral_4	139	88.97
numeral_5	7	66.67
numeral_6	26	70.00
numeral_7	14	0.00
numeral_8	29	60.00
ornament(s)	75	91.00
other_text	271	72.01
other-dot	197	72.65
quarter_rest	803	95.83
repeat	84	73.99
repeat-dot	266	94.52
sharp	2068	99.28
slur	2602	86.43
staccato-dot	1385	93.38
staff_grouping	191	NaN
stem	21417	98.52
tempo_text	116	52.91
tenuto	146	75.91
thin_barline	3332	98.98
tie	704	81.97
time_signature	192	95.56
trill	179	79.53
tuple	244	74.58
tuple_bracket/line	51	NaN
whole_rest	153	96.07

[19] B. Coüasnon, "Dmos: a generic document recognition method, application to an automatic generator of musical scores, mathematical formulae and table structures recognition systems," in *Proceedings of Sixth International Conference on Document Analysis and Recognition*, 2001, pp. 215–220.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, *Microsoft COCO: Common Objects in Context*. Cham: Springer International Publishing, 2014, pp. 740–755.

[21] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 248–255.

[22] J. j. Hajič and P. Pecina, "The MUSCIMA++ dataset for handwritten optical music recognition." *Proceedings of the 14th IAPR International Conference on Document Analysis and Recognition*, 2017.

[23] A. Fornés, A. Dutta, A. Gordo, and J. Lladós, "CVC-MUSCIMA: a ground truth of handwritten music score images for writer identification and staff removal," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 15, no. 3, pp. 243–251, 2012.

[24] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, #jan# 2015.

[25] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *CoRR*, vol. abs/1708.02002, 2017.

[26] A. Fornés, A. Dutta, A. Gordo, and J. Lladós, *The 2012 Music Scores Competitions: Staff Removal and Writer Identification*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 173–186.